Robotica (2016) volume 34, pp. 634–651. © Cambridge University Press 2014
doi:10.1017/S0263574714001775

PSO–Lyapunov motion/force control of robot arms with model uncertainties

Haifa Mehdi* and Olfa Boubaker†

National Institute of Applied Sciences and Technology, INSAT, Centre Urbain Nord, BP 676-1080 Tunis Cedex, Tunisia

(Accepted June 6, 2014. First published online: July 4, 2014)

SUMMARY
A method for motion/force control of robot arms with model uncertainties is presented. Tracking control of complex trajectories is guaranteed using a Lyapunov approach with high-precision performance ensured using a particle swarm optimization (PSO) algorithm. Tracking performance and robustness are simulated for a robotic device for limb rehabilitation that is designed to be adapted easily to different subjects by considering model parameter uncertainties. Controller parameters are optimized offline using the PSO algorithm with Lyapunov stability conditions considered as inequality constraints. Using the control scheme, the robot can guide limbs on smooth and non-smooth trajectories, under model uncertainties and measurement noise.

KEYWORDS: Motion/force control; Stiffness control; Constrained motion; Robot arm; Lyapunov theory; Model uncertainties.

1. Introduction

Motion/force control of robotic arms in constrained environments has up to now been considered as the most complex and challenging problem in the field of robotics.[1] The handbook by Siciliano and Khatib,[1] the books by Siciliano and Villani,[2] and Carbone,[3] and the surveys by Zeng and Hemami,[4] Chiaverini *et al*.,[5] Briones *et al*.,[6] and Carbone *et al*.[7] explain the complexity of this control problem.

When a robot is in constrained motion, unstable behavior can occur, with risks to the environment and/or the robot. Indeed, the control of constrained robotic systems is a multivariate and inherently nonlinear control problem with conflicting objectives. In fact, any position eliminates the associated force in a direction of the operational space, and any force eliminates the associated position in other directions of the operational space. Therefore, finding stability conditions remains a challenging problem. In addition, fine tuning the controller parameters of robotic arms to maximize their capabilities remains a tricky dilemma because the optimization procedure is very expensive and time-consuming.[8] The problem becomes even more difficult when uncertainties are present. Uncertainties can arise in the constrained environment, for example, a surface with unknown geometrical characteristics,[9,10] or an unknown impedance or stiffness.[11,12] They can also arise in the robotic model, for example, unknown robotic parameters[13,14] or uncertainties in the Jacobian matrix.[15]

To optimize the controller parameters of robotic systems, particle swarm optimization (PSO) techniques have been widely used. Recently, PSO has attracted much attention in many areas. The books by Bonabeau *et al*.[16] and Kennedy *et al*.[17] and the surveys by Clerc and Kennedy,[18] Dorigo and Birattari,[19] Bonabeau *et al*.,[20] and Dorigo *et al*.[21] reveal the richness of this research field. The most important aspect of PSO compared with other evolutionary algorithms is its faster convergence speed and simpler implementation.[22]

* Corresponding author. E-mail: haifa.mehdi@gmail.com, olfa.boubaker@insat.rnu.tn

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Much work has been done over many years on the development of dynamic controllers based on PSO. However, in the robotics field, most research using PSO has been devoted to solving robotic swarming problems,[23–25] motion planning problems, and navigation and obstacle-avoidance problems.[26–28] Only a few papers have been published on PSO in the context of motion control of robotic arms in non-constrained environments,[29–31] and we have also noted a lack of results for motion/force control in constrained environments.[32]

On the other hand, robotic arms have also been the subject of many years of research, and the force and motion tracking control of robot arms has been widely applied in many areas such as industrial production,[33] space robotics,[34] underwater vehicles,[35] and rehabilitation robotics.[36] Unlike space and underwater robotics, where 5 to 7 degrees of freedom (DOF) are generally used in very constrained situations in three-dimensional spaces, and for which control accuracy and speed are the main challenges, devices for aiding rehabilitation must have only 2 or 3 DOF with planar motion for greater therapeutic efficiency.[36–39] Such robotic devices must provide sufficient safety by allowing the robotic device to be adaptable to different patients and to support their progress during therapy. The tracked trajectories must be complex, such as circular ones, but not very aggressive like those imposed in space or underwater robotics.

The most important contribution of this paper is the proposal of a new design of motion/force control of robot arms with model uncertainties. New sufficient conditions are proposed for safe tracking of complex trajectories using a Lyapunov approach in which high-precision performance is ensured using a PSO algorithm. The tracking performance and robustness of the proposed approach are tested, by means of numerical simulations, on a robot with 3 DOF for rehabilitation of upper limbs. The controller parameters are optimized offline using a PSO algorithm in which the new Lyapunov stability conditions are considered as inequality constraints for the optimization problem.

The paper is organized as follows. In Section 2, the problem is stated. New stability conditions using Lyapunov theory are given for the constrained robotic arm with model uncertainties in Section 3. In Section 4, the PSO algorithm is described and is used to optimize the controller parameters. Finally, in Section 5, the controller parameters are optimized and the simulation results are performed on an aided-rehabilitation robotic system.

2. Problem Statement

Consider a constrained robotic manipulator with n DOF described by the following uncertain dynamical model:

$$[M(\theta) + \Delta M]\ddot{\theta} + H(\theta, \dot{\theta}) + \Delta H + G + \Delta G = U - J^T F, \tag{1}$$

where $\theta, \dot{\theta}, \ddot{\theta} \in R^n$ are joint position, joint velocity, and joint acceleration vectors, respectively. $M(\theta) \in R^{n\times n}$, $H(\theta, \dot{\theta}) \in R^n$, and $G(\theta) \in R^n$ denote the inertia matrix, the vector of centrifugal and Coriolis forces, and the vector of gravity terms, respectively, and $\Delta M \in R^{n\times n}$, $\Delta H \in R^{n\times 1}$, and $\Delta G \in R^{n\times 1}$ denote the respective uncertainties. $U \in R^n$ is the control vector, $F \in R^p$ is the vector of contact forces exerted by the robot on the environment (where the task space dimension is p), and $J(\theta) \in R^{p\times n}$ is the Jacobian matrix.

The forward kinematic model of the robot is described by

$$X = h(\theta), \tag{2}$$

and the differential kinematic model by

$$\dot{X} = \frac{\partial h(\theta)}{\partial \theta} = J(\theta)\dot{\theta}, \tag{3}$$

where $X, \dot{X} \in R^p$ denote the Cartesian position vector and the Cartesian velocity vector, respectively. $h(\theta) : R^n \to R^p$ is a vector of nonlinear functions.

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Given a desired Cartesian position $X_d \in R^p$ of the end-effector of the robotic manipulator and an end time t_f, the control problem aims to ensure that

$$\lim_{t \to t_f} X_d - X = 0$$

under the following assumptions:

A_1: The vectors of joint position θ and joint velocity $\dot{\theta}$ are measured.
A_2: The matrix $J(\theta)$ is of full rank and bounded for all $\theta \in R^n$.
A_3: M and ΔM are symmetric positive matrices.
A_4: At $t = 0$, $\Delta M = 0$, $\Delta H = 0$, and $\Delta G = 0$.
A_5: The vector of uncertainties related to the gravity vector is assumed to be bounded such that $\|\Delta G\| \leq \alpha \|\dot{\theta}\|$, where α is a positive constant.
A_6: All feedback gain matrices used to solve the control problem are diagonal.

In the following, we will adopt the following notation:

$$M(\theta) = M,\ H(\theta, \dot{\theta}) = H,\ G(\theta) = G,\ J(\theta) = J.$$

3. Robust Control-Based Lyapunov Approach

Theorem: The constrained robotic system described by the uncertain dynamical model

$$(M + \Delta M)\ddot{\theta} + H + \Delta H + G + \Delta G = U - J^T F \tag{4}$$

is asymptotically stable for the contact force described by

$$F = K_e(X_d - X), \tag{5}$$

where $K_e \in R^{p \times p}$ is the environment stiffness, under the robust control law described by

$$U = MJ^{-1}[K_p(X_d - X) + K_v(\dot{X}_d - \dot{X}) + F] + J^T F + J^{-1}(\dot{X}_d - \dot{X})K\sigma\ \text{sign}(\sigma) + G, \tag{6}$$

where σ is a nonlinear function defined by

$$\sigma = C[(\dot{X}_d - \dot{X}) + \Lambda(X_d - X)], \tag{7}$$

for a positive constant gain K, a constant vector $C \in R^{\ell \times p}$, and a positive constant matrix $\Lambda \in R^{p \times p}$, if there exist diagonal gain matrices K_p, $K_v \in R^{p \times p}$ satisfying the following conditions:

$$\begin{cases} K_p + K_e > 0, \\ K_v > 0. \end{cases} \tag{8}$$

Proof: Considering the uncertain robotic system (4) for the force model (5) and the control law (6), we have

$$\begin{aligned} &(M + \Delta M)\ddot{\theta} + H + \Delta H + \Delta G - MJ^{-1}[(K_p + K_e)(X_d - X) + K_v(\dot{X}_d - \dot{X})] \\ &\quad - J^{-1}(\dot{X}_d - \dot{X})\,K\sigma\ \text{sign}(\sigma) = 0, \end{aligned} \tag{9}$$

and, denoting by $\Phi = \theta - \theta_d$ and $Y(\Phi) = X(\theta) - X_d$ the error vectors defined in the joint and task spaces, respectively, we can write this as

$$\begin{aligned} &[M(\Phi) + \Delta M]\ddot{\Phi} + H(\Phi, \dot{\Phi}) + \Delta H + \Delta G + M(\Phi)J^{-1}(\Phi)[(K_p + K_e)Y(\Phi) + K_v\dot{Y}(\Phi)] \\ &\quad + J^{-1}(\Phi)\dot{Y}(\Phi)\,K\sigma\ \text{sign}(\sigma) = 0. \end{aligned} \tag{10}$$

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Following the same development as presented in ref. [36] for a kinetic energy defined by

$$T(\Phi, \dot{\Phi}) = \frac{1}{2}\dot{\Phi}^T[M(\Phi) + \Delta M]\dot{\Phi}, \tag{11}$$

and taking into account that $M(\Phi)$ and ΔM are symmetric matrices, we can prove that

$$H(\Phi, \dot{\Phi}) + \Delta H = \sum_{i=1}^{n}\left(\frac{\mathrm{d}\Phi_i}{\mathrm{d}t}\frac{\partial[M(\Phi) + \Delta M]}{\partial \Phi_i}\right)\frac{\dot{\Phi}}{2}, \tag{12}$$

$$\frac{\partial P(\Phi)}{\partial \Phi} = M(\Phi)J^{-1}(\Phi)(K_p + K_e)Y(\Phi), \tag{13}$$

$$\frac{\partial D(\Phi, \dot{\Phi})}{\partial \dot{\Phi}} = M(\Phi)J^{-1}(\Phi)K_v\dot{Y}(\Phi) + J^{-1}(\Phi)\dot{Y}(\Phi)K\sigma\ \mathrm{sign}(\sigma) + \Delta G. \tag{14}$$

We now impose on system (10) a Lyapunov Hamiltonian function defined by[39]

$$V(\Phi, \dot{\Phi}) = T(\Phi, \dot{\Phi}) + P(\Phi) - P(0). \tag{15}$$

Again following the same development as presented in ref. [36], the first Lyapunov condition is proved, since $M(0) = 0,\ \Delta M(0) = 0$, and

$$\left[\frac{\partial V_p(\Phi)}{\partial \Phi}\right]_{\Phi=0} = M(0)J^{-1}(0)(K_p + K_e)Y(0) = 0, \tag{16}$$

and the second Lyapunov condition will be then satisfied if $K_p + K_e$ is positive-definite, since $M(\Phi)$ is defined as a positive matrix (assumption A_3).

To prove the third Lyapunov condition, let us give the derivative of Lyapunov function (16):

$$\frac{\mathrm{d}V(\Phi)}{\mathrm{d}t} = \frac{\mathrm{d}T(\Phi, \dot{\Phi})}{\mathrm{d}t} + \frac{\mathrm{d}P(\Phi)}{\mathrm{d}t}. \tag{17}$$

From (11) and (13), we can write

$$\begin{aligned}\frac{\mathrm{d}T(\Phi, \dot{\Phi})}{\mathrm{d}t} &= \dot{\Phi}^T[M(\Phi) + \Delta M]\ddot{\Phi} + \dot{\Phi}^T\frac{\mathrm{d}[M(\Phi) + \Delta M]}{\mathrm{d}t}\frac{\dot{\Phi}}{2} \\ &= \dot{\Phi}^T[M(\Phi) + \Delta M]\ddot{\Phi} + \dot{\Phi}^T[H(\Phi, \dot{\Phi}) + \Delta H].\end{aligned} \tag{18}$$

From (13), we can write

$$\begin{aligned}\frac{\mathrm{d}P(\Phi)}{\mathrm{d}t} &= \dot{\Phi}^T\frac{\partial P(\Phi)}{\partial \Phi} \\ &= \dot{\Phi}^T M(\Phi)J^{-1}(\Phi)(K_p + K_e)Y(\Phi).\end{aligned} \tag{19}$$

Using (18) and (19), we can write

$$\frac{\mathrm{d}V(\Phi, \dot{\Phi})}{\mathrm{d}t} = \dot{\Phi}^T[M(\Phi) + \Delta M]\ddot{\Phi} + \dot{\Phi}^T[H(\Phi, \dot{\Phi}) + \Delta H] + \dot{\Phi}^T M(\Phi)J^{-1}(\Phi)(K_p + K_e)Y(\Phi). \tag{20}$$

From (10), we can write

$$\begin{aligned}&[M(\Phi) + \Delta M]\ddot{\Phi} + [H(\Phi, \dot{\Phi}) + \Delta H] + M(\Phi)J^{-1}(\Phi)(K_p + K_e)Y(\Phi) \\ &\qquad = -\Delta G - M(\Phi)J^{-1}(\Phi)K_v\dot{Y}(\Phi) - J^{-1}(\Phi)\dot{Y}(\Phi)K\sigma\ \mathrm{sign}(\sigma).\end{aligned} \tag{21}$$

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Using (21), we obtain from (20)

$$\frac{\mathrm{d}V(\Phi, \dot{\Phi})}{\mathrm{d}t} = -\dot{\Phi}^T M(\Phi) J^{-1}(\Phi) K_v \dot{Y}(\Phi) - \dot{\Phi}^T J^{-1}(\Phi) \dot{Y}(\Phi) K \sigma \operatorname{sign}(\sigma) - \dot{\Phi}^T \Delta G. \tag{22}$$

With the kinematic model for the error dynamics, (22) gives

$$\begin{aligned}\frac{\mathrm{d}V(\Phi, \dot{\Phi})}{\mathrm{d}t} = &-\dot{Y}^T(\Phi) J^{-T}(\Phi) M(\Phi) J^{-1}(\Phi) K_v \dot{Y}(\Phi) \\ &- \dot{Y}^T(\Phi)(J J^T)^{-1}(\Phi) \dot{Y}(\Phi) K \sigma \operatorname{sign}(\sigma) - \dot{\Phi}^T \Delta G.\end{aligned} \tag{23}$$

By assumption A_5, the third Lyapunov condition is satisfied if the matrices K_v and K are both positive-definite.

4. Particle Swarm Controller Optimization

PSO is a stochastic population-based optimization approach inspired by the study of swarms. It was introduced by Kennedy and Eberhart in 1995.[40] In the most common implementations of PSO, particles move through the search space using a combination of an attraction to the best solution that they have individually found and an attraction to the best solution that any particle in their neighborhood has found. Neighborhoods unconnected to the locality of the particle have come into use, including what has come to be known as a global neighborhood, or *g*best model, where each particle is connected to and able to obtain information from every other particle in the swarm.[41] The PSO concept consists of changing the velocity of each particle i toward its best position, *p*best, at each time step. For the ith particle, the new velocity and new position are updated such that for $k = 1,\ 2,\ \ldots,\ N$ and $i = 1,\ 2,\ \ldots,\ M$,[42]

$$V_i^{(k+1)} = w^{(k)} V_i^{(k)} + c_1 \operatorname{rand}(.) \left(p\mathrm{best}_i^{(k)} - p_i^{(k)} \right) + c_2 \operatorname{Rand}(.) \left(g\mathrm{best}^{(k)} - p_i^{(k)} \right), \tag{24}$$

$$p_i^{(k+1)} = p_i^{(k)} + V_i^{(k+1)}, \tag{25}$$

where

M = number of particles in a group;
N = maximum number of iterations;
$V_i^{(k)}$ = velocity of ith particle at iteration k;
$p_i^{(k)}$ = position of ith particle at iteration k;
$p\mathrm{best}_i^{(k)}$ = best position of ith particle at iteration k;
$g\mathrm{best}^{(k)}$ = best position of the group until iteration k;
$w^{(k)}$ = inertia weight factor at iteration k;
c_1, c_2 = acceleration factors;
$\operatorname{rand}(.),\ \operatorname{Rand}(.)$ = random numbers between 0 and 1.

In general, the inertia weight is calculated according to the following equation:

$$w^{(k)} = k \frac{w_{\max} - w_{\min}}{N}. \tag{26}$$

The design steps for implementing the PSO algorithm are given in Fig. 1.

Fig. 2 is a block diagram of the robust control approach tuned with PSO for the constrained robot manipulator.

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

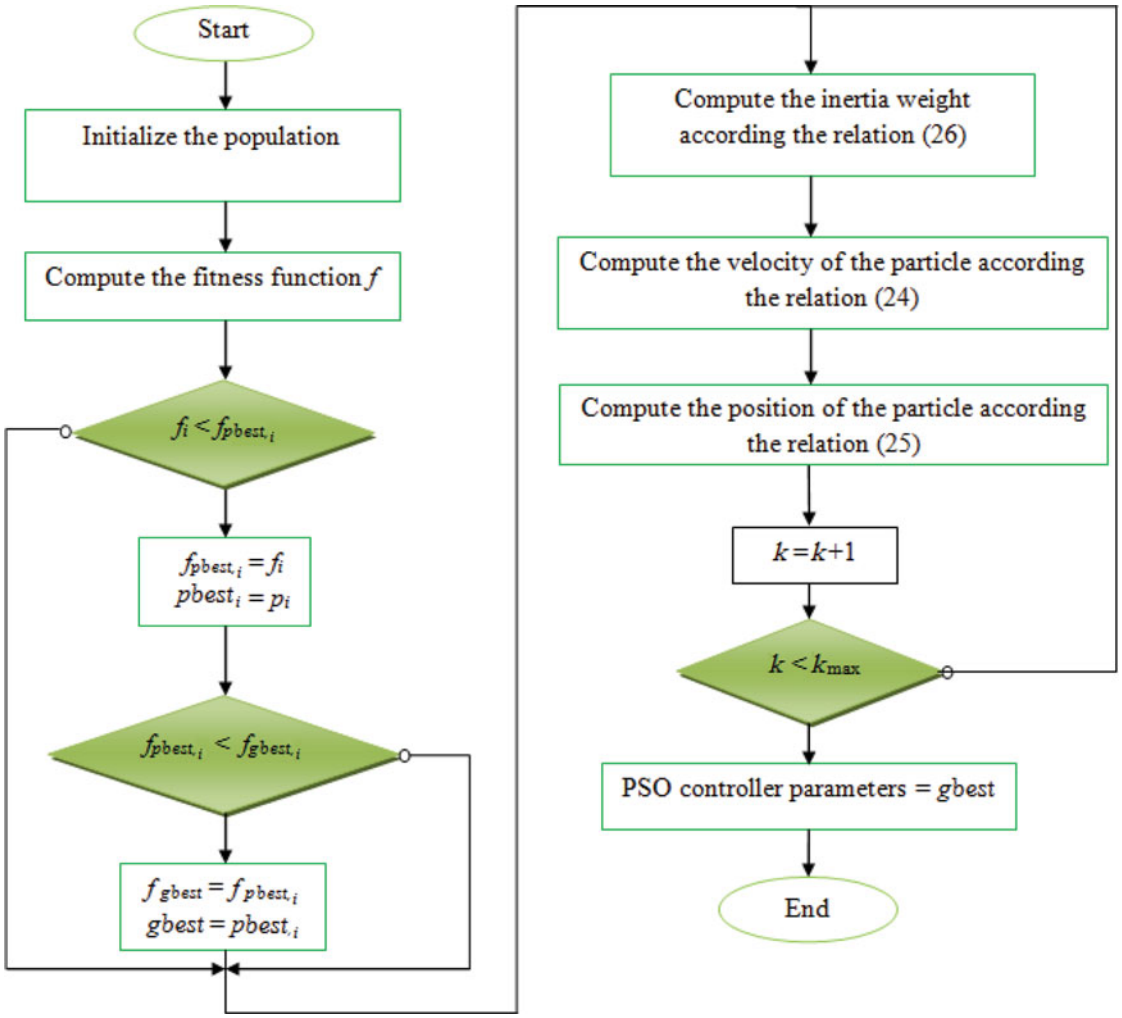


Fig. 1. Controller parameter tuning via PSO.

5. Application

In this section, a robot-aided rehabilitation device[36] will be used as a case study to prove the efficiency of the proposed approach. Such robotic devices are used by therapists as part of the program of upper or lower limb rehabilitation in stroke patients.

5.1. The robot-aided rehabilitation device

The robot-aided rehabilitation device designed for upper limb rehabilitation is composed of three parts (Fig. 3): a 3-DOF planar robot (Fig. 4), two brackets maintaining the upper limb of the patient over the robotic device, and finally a linkage system allowing the therapist to customize the device for different users. The 3-DOF robot must mimic, in a planar space, the motion of the shoulder, elbow, and wrist joints of the patient and must be configured for safe compliant motion by allowing the application of corrective forces and torques to the human arm. The desired tracking trajectories are then imposed by the robotic system.

5.2. The constrained robotic model

The different matrices related to the dynamical model (1), the kinematic model (2), and the differential kinematic model (3) are defined by[36]

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

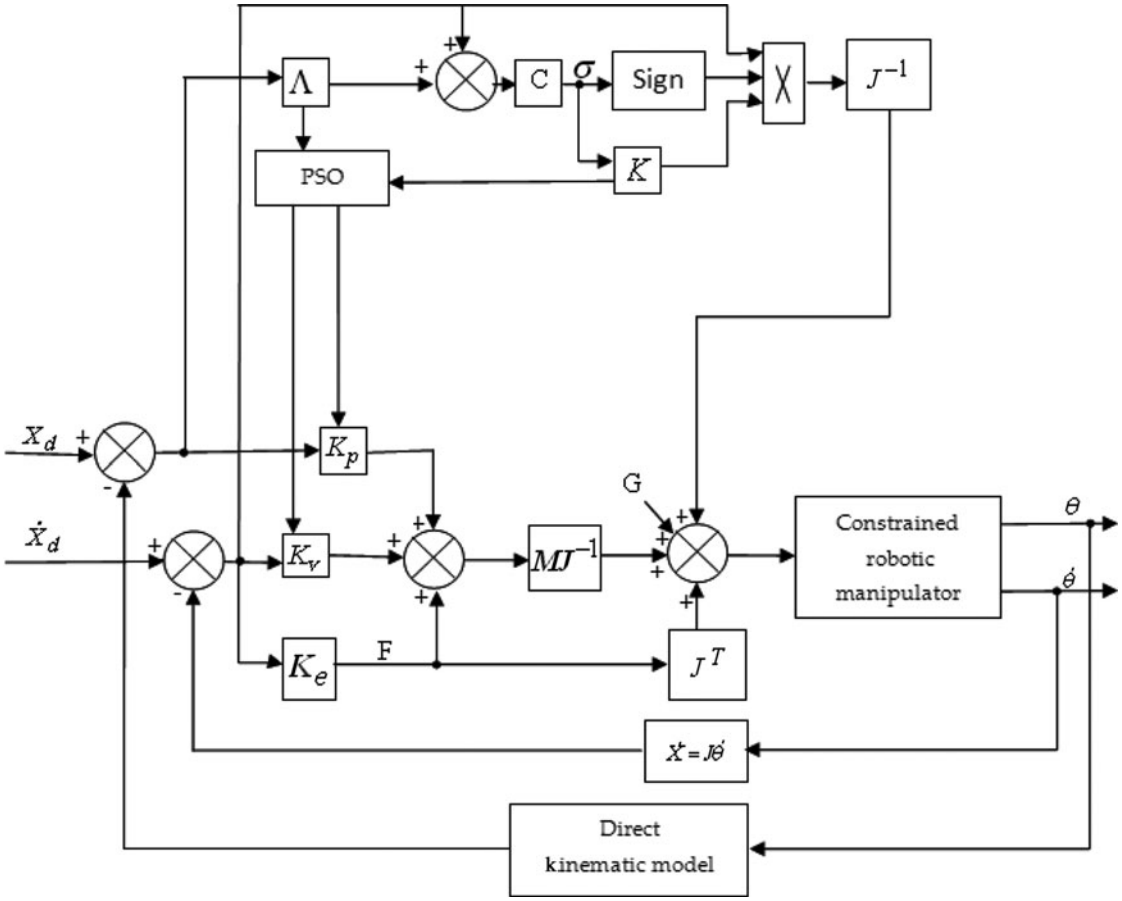


Fig. 2. Block diagram of the robust control approach tuned with PSO.

$$h(\theta) = \begin{bmatrix} \sum_{i=1}^{3} L_i \cos\theta_i \\ \sum_{i=1}^{3} L_i \sin\theta_i \end{bmatrix},$$

$$J(\theta) = \begin{bmatrix} -L_1\sin\theta_1 & -L_2\sin\theta_2 & -L_3\sin\theta_3 \\ L_1\cos\theta_1 & L_2\cos\theta_2 & L_3\cos\theta_3 \end{bmatrix},$$

$$M(\theta) = \begin{bmatrix} I_1 + m_1k_1^2 + m_2L_1^2 + m_3L_1^2 & (m_2L_1k_2 + m_3L_1L_2)\cos(\theta_1-\theta_2) & m_3L_1k_2\cos(\theta_1-\theta_3) \\ (m_2L_1k_2 + m_3L_1L_2)\cos(\theta_1-\theta_2) & I_2 + m_2k_2^2 + m_3L_2^2 & m_3L_2k_3\cos(\theta_2-\theta_3) \\ m_3L_1k_2\cos(\theta_1-\theta_3) & m_3L_2k_3\cos(\theta_2-\theta_3) & I_3 + m_3k_3^2 \end{bmatrix},$$

$$H(\theta,\dot{\theta}) = \begin{bmatrix} 0 & (m_2L_1k_2 + m_3L_1L_2)\sin(\theta_1-\theta_2) & m_3L_1k_3\sin(\theta_1-\theta_3) \\ -(m_2L_1k_2 + m_3L_1L_2)\sin(\theta_1-\theta_2) & 0 & m_3L_2k_3\sin(\theta_2-\theta_3) \\ -m_3L_1k_3\sin(\theta_1-\theta_3) & -m_3L_2k_3\sin(\theta_2-\theta_3) & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix},$$

$$G(\theta) = g \begin{bmatrix} (m_1k_1 + m_2L_1 + m_3L_1)\cos\theta_1 \\ (m_2k_2 + m_3L_2)\cos\theta_2 \\ m_2k_3\cos\theta_3 \end{bmatrix},$$

where m_i, L_i, k_i, and I_i $(i = 1,\ 2,\ 3)$ are mass, length, position of center of gravity, and inertia parameters. The numerical values of these physical parameters of the robot are given in Table I.[36]

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Table I. Physical parameters of the robotic arm.

Link i	m_i (kg)	L_i (m)	k_i (m)	I_i (kg m^{-2})
1	1.960	0.321	0.140	0.016
2	1.120	0.253	0.109	0.006
3	0.420	0.187	0.095	0.001

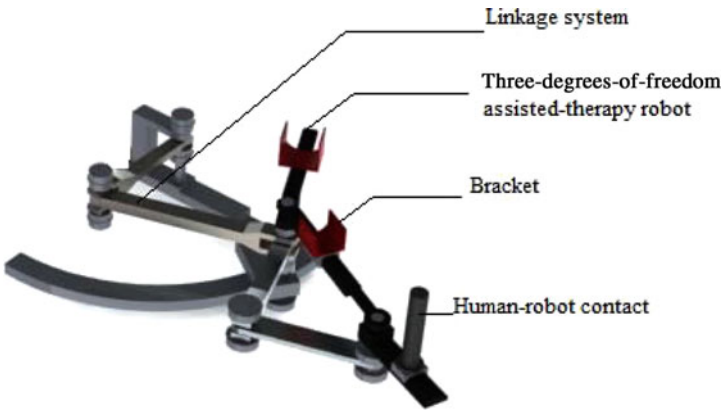


Fig. 3. The robot-aided rehabilitation device.

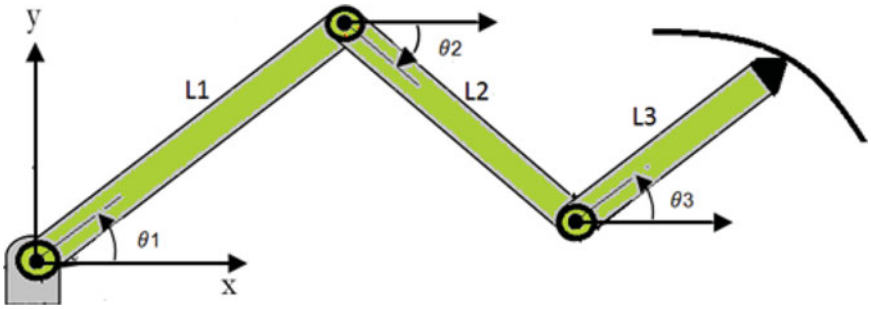


Fig. 4. 3 DOF constrained robot.

5.3. The constrained optimization problem via PSO

A fundamental step in applying the PSO algorithm is the choice of the cost used to evaluate the fitness function of each particle. In this paper, we propose to use the mean root squared error cost function defined in the task space of the robot by

$$\text{MRSE} = \frac{1}{N}\left(\sum_{k=1}^{N} \{[X_d(k) - X(k)]^T [X_d(k) - X(k)]\}^{1/2}\right). \tag{27}$$

Let the stiffness vector be defined by its components in Cartesian space,

$$K_e = \text{diag}[K_{e,x}\ K_{e,y}]^T,$$

and let the decision vector be defined by

$$z = [k_{p_1}\ k_{p_2}\ k_{v_1}\ k_{v_2}\ \Lambda_1\ \Lambda_2\ K]^T.$$

The following dynamic constrained optimization problem is then solved via PSO:

Minimize MRSE

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

subject to:

$$M(\theta)\ddot{\theta} + H(\theta, \dot{\theta}) + G(\theta) = U - J(\theta)^T F, \tag{28}$$

$$\dot{X} = J(\theta)\dot{\theta}, \tag{29}$$

$$I_{(7\times 7)}z > [-k_{e,x} - k_{e,y}\, 0\, 0\, 0\, 0\, 0]^T, \tag{30}$$

$$1 \leq K \leq 5, \tag{31}$$

and, for $i = 1, 2$,

$$100 \leq k_{pi} \leq 900, \tag{32}$$

$$30 \leq k_{vi} \leq 500, \tag{33}$$

$$1 \leq \Lambda_i \leq 10. \tag{34}$$

The relations (28) and (29) represent the dynamic and kinematic constraints of the optimization problem, respectively, whereas the inequality constraint (30) is equivalent to the stability condition (8). The inequalities (31)–(34) define a set of lower and upper bounds of the decision variables chosen with regard to practical implementation.

5.4. Case studies

To explore the performance of the proposed approach, we consider two different tasks for simulation.

Task 1: In the first task, the robot is controlled to track a given smooth circular profile in the task space with a radius of 0.76 m. Furthermore, we select a cubic polynomial function in the joint space beginning at $\theta_{i0} = [0 \quad 0 \quad 0]^T$ and ending at $\theta_{id} = [\pi \quad \pi \quad \pi]^T$ during 1 s. The circular trajectory is described in ref. [36]. Three different cases are explored for this particular task. The objective is to analyze the influences of uncertainties and noise measurement on the control law profiles and the tracking performances.

1. **Case 1:** Parameter uncertainties affecting the constrained robotic model are considered here. They are described by

$$\widehat{m}_1 = 1.96 + 0.1 \sin t, \widehat{m}_2 = 1.12 + 0.05 \sin t, \widehat{m}_3 = 0.42 + 0.3 \sin t.$$

   It can be shown that, on the dynamic model of the constrained robotic model, the parameter uncertainties affect $M(\theta)$, $H(\theta, \dot{\theta})$, and also $G(\theta)$. They consequently influence the value of the parameter α given by assumption A_5.
2. **Case 2:** In addition to the model uncertainties, sensory noises are introduced in the joint position and joint velocity measurements: we consider Gaussian noise of 0.2° mean and 0.2° standard deviation for the joint position measurements and 0.05° s^{-1} mean and 0.05° s^{-1} standard deviation for the joint velocity measurements.
3. **Case 3:** The Gaussian noise of the sensory joint position is maintained at 0.2° mean and 0.2° standard deviation, whereas the Gaussian noise for the joint velocity measurements is increased to 0.2° s^{-1} mean and 0.2° s^{-1} standard deviation. The goal here is to analyze the effect of the noise in the velocity measurements compared with the noise in the position measurements.

Task 2: The robot-aided rehabilitation is controlled to track a non-smooth trajectory. It must be noted here that such an aggressive movement is not employed in the first stages of treatment of injured limbs, but can be beneficial in the final stage of rehabilitation. We choose the non-smooth trajectory here to have a rectangular profile[43] 1 m in width and 0.4 m in height. Furthermore, we select a cubic polynomial function in the joint space beginning at $\theta_{i0} = [0 \quad 0 \quad 0]^T$ and ending at $\theta_{id} = [2\pi \quad 2\pi \quad 2\pi]^T$ during 7 s. Two cases will be explored here to identify the influence of uncertainties on tracking performances and on control law profiles for non-smooth trajectories.

1. **Case 1:** Only the mass uncertainties, described above, are considered in this case.
2. **Case 2:** In addition to the mass uncertainties, we assume that the therapist increases the lengths of the three links of the robot by 20% at $t = 3.5$ s to provide better comfort for the patient.

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Table II. Parameters of PSO algorithm.

Designation	Variable	Value
Number of particles in a group	M	30
Maximum number of iterations	N	40
Minimum inertia weight factor	$w_{\min}$	0.4
Maximum inertia weight factor	$w_{\max}$	0.9
Search interval for $k_{p,i}$	$[k_{p_{\min}}\ k_{p_{\max}}]$	[100 900]
Search interval for $k_{v,i}$	$[k_{v_{\min}}\ k_{v_{\max}}]$	[30 500]
Search interval for Λ_i	$[\Lambda_{\min}\ \Lambda_{\max}]$	[1 10]
Search interval for K	$[k_{\min}\ k_{\max}]$	[1 5]
Search interval velocity of $k_{p,i}$	$[0\ k_{p_{\max}}/2]$	[0 450]
Search interval velocity of $k_{v,i}$	$[0\ k_{v_{\max}}/2]$	[0 250]
Search interval velocity of Λ_i	$[0\ \Lambda_{\max}/2]$	[0 5]
Search interval velocity of K	$[0\ k_{\max}/2]$	[0 2.5]
Acceleration constants	$c_1,\ c_2$	2

Table III. Best PSO controller parameters.

		k_{p_1}	k_{p_2}	k_{v_1}	k_{v_2}	Λ_i	K
Task 1	Case 1	173.2	690.2	266.5	334.4	9.3	4.2
	Case 2	597.8	800	347.1	390.3	7.5	2
	Case 3	718	790.6	157.6	400	8.1	3.7
Task 2	Case 1	900	850.2	380	400	6	1.8
	Case 2	830.2	900	146.3	264	5.8	2.1

5.5. Simulation results and performance analysis

The constrained optimization problem is solved for $K_e = \text{diag}\,[10 \quad 10]$ and for the initial decision vector $z_0 = [200\,450\,300\,250\,5\,7\,2]^T$, and the parameters of the PSO algorithm are given in Table II. Table III gives the best PSO controller parameters for the different tasks and for the different cases of each task.

Task 1: The evolution of the fitness function for case 1 of task 1 is shown in Fig. 5. As can be seen, the PSO algorithm is run once. This means that in contrast to most control laws based on PSO, where the optimizer must tune the controller parameters dynamically and add the control action directly, in the approach proposed in this paper, the controller parameters have to be optimized offline. The major advantage of such an approach is that in practice there are no problems with control law implementation when the optimizer and the controller operate at different speeds.

The desired and actual trajectories are plotted in Figs. 6(a), 7(a), and 8(a) for cases 1, 2, and 3, respectively, of task 1. These simulation results prove the stability, high precision, and robustness of the tracking control approach in the presence of model uncertainties and sensory noise. The position and velocity errors in Cartesian space are shown for cases 1, 2, and 3 in Figs. 6(b), 7(b), and 8(b), respectively. From the results obtained for cases 2 and 3, we can conclude that noise is translated into position errors and velocity errors. Figs. 6(c), 7(c), and 8(c) show the profiles of the constrained forces and of the control laws for cases 1, 2, and 3, respectively. Comparing the results obtained for cases 2 and 3, we can conclude that noise in velocities has a greater influence on the degradation of controller performance than noise in position measurements. The effects of vibration are observed in control laws when the velocity noise is increased. However, it is clear that even if the noise is not modeled in the design of the controller, the proposed approach ensures robustness and tracking performance.

Task 2: Results from computer simulations for the two cases of task 2 are plotted in Figs. 9 and 10. The desired and actual trajectories are shown in Figs. 9(a) and 10(a) for cases 1 and 2, respectively. We can confirm that the desired trajectory was followed completely by the robotic device in the presence of mass and length uncertainties, even though the tracking trajectory was non-smooth. The position and velocity errors in Cartesian space are shown in Figs. 9(b) and 10(b) for cases 1 and 2, respectively. In the case of mass uncertainties, position errors are more important for task 2 than

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

Table IV. Model uncertainties and sensory noise effects on the tracking trajectory and the controller dynamics for different constrained trajectories.

Tasks	Cases	α	Best fitness	Processing time (s)	Velocity (m s^{-1})
Task 1	Case 1	0.95	2.3×10^{-4}	439	1.42
	Case 2	0.8	7×10^{-5}	414	1.04
	Case 3	0.97	3.4×10^{-5}	500	0.96
Task 2	Case 1	1.43	5.6×10^{-4}	830	0.18
	Case 2	1.75	7.2×10^{-4}	1027	1.22

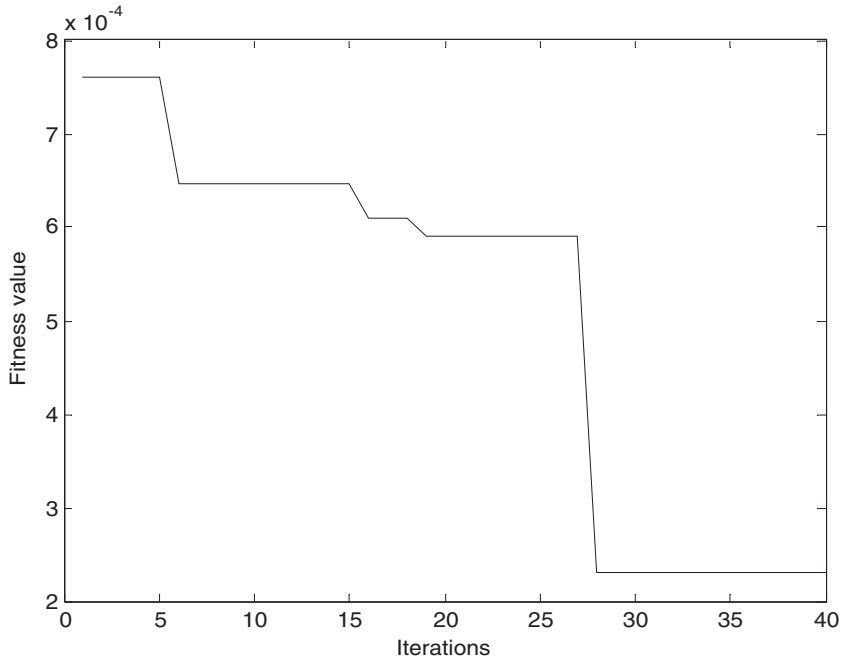


Fig. 5. Evolution of the fitness function for case 1 of task 1.

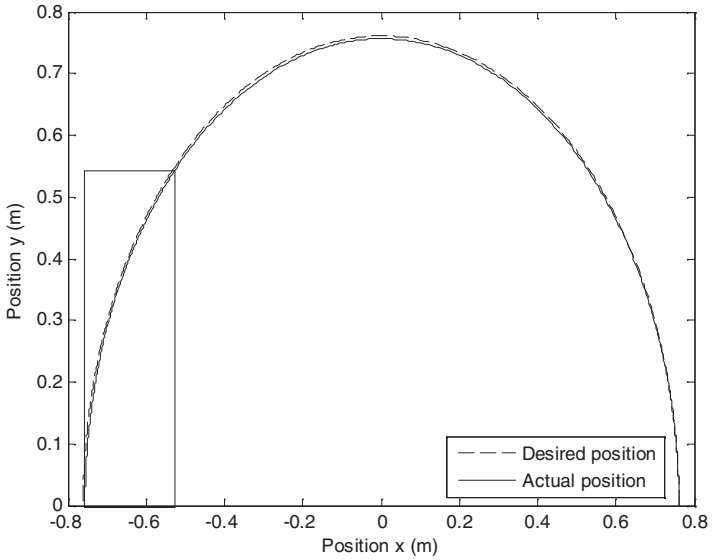
for task 1. These errors increase further with the introduction of length uncertainties. Furthermore, more overshoots in velocity are observed. From Figs. 9(c) and 10(c), which show the profiles of the constrained forces and the controller laws for cases 1 and 2, respectively, we can conclude that similar overshoots are observed in the force profiles, whereas the profile of the controller laws remains smooth. Such observations prove that the proposed approach performs well in adjusting the control laws for non-smooth motions even if uncertainties are present in different parameters.

Table IV presents the effect of model uncertainties and noise measurement on tracking performance. First, these results prove that the parameter α, given by assumption A_5 and computed as $\|\Delta G\|/\|\dot{\theta}\|$, is very sensitive to these perturbations. On the other hand, we must remark that the fitness function always has the same range of values for all tasks and all cases. This observation confirms the high tracking performance of the control laws even in critical situations. We can also conclude that the processing time is influenced mainly by the smoothness of the tracking trajectory and the uncertainties. It is very sensitive to velocity noise. Finally, the velocity of the robotic arm is improved for smoother trajectories

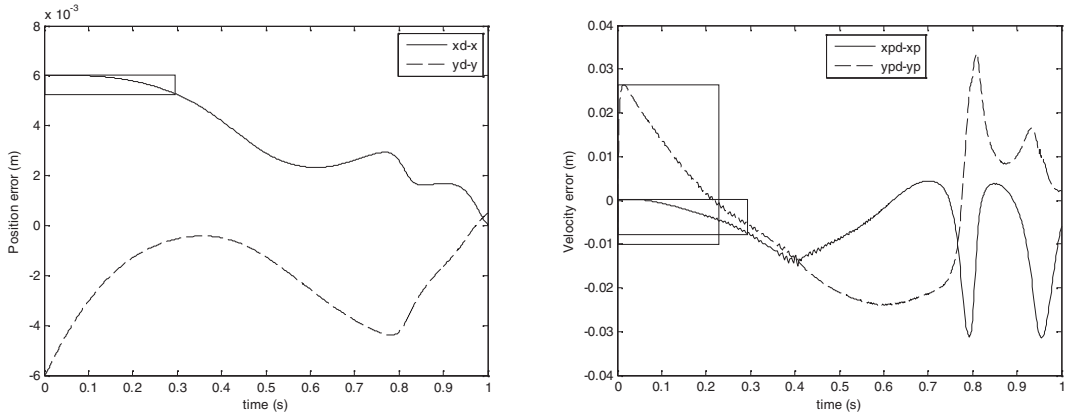
6. Conclusion

This paper has proposed a new design of robust motion/force control for robotic arms with model uncertainties. Robust asymptotic stability conditions have been shown using Lyapunov theory, with tracking performances being ensured via particle swarm intelligence, which was used to optimize,

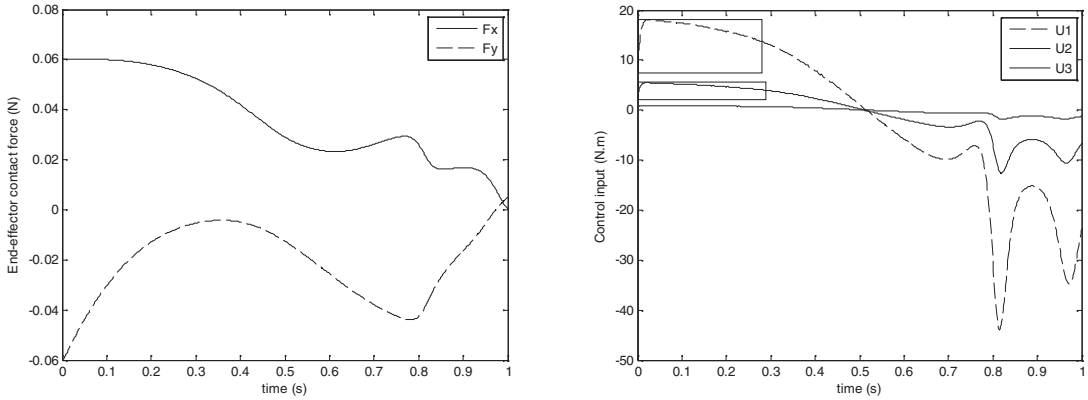
https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press



(a)



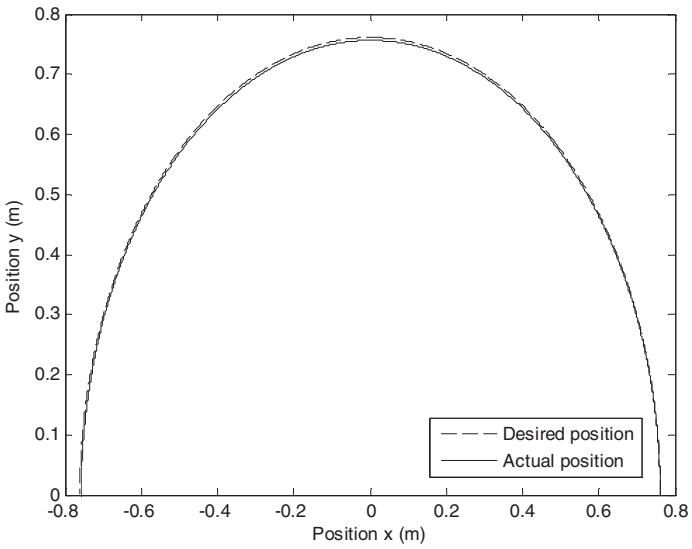
(b)



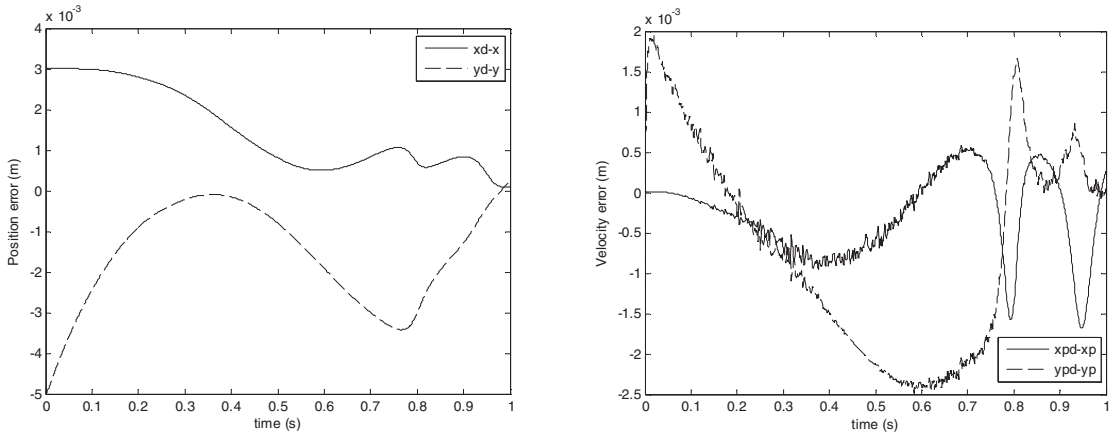
(c)

Fig. 6. Tracking dynamics of the robot for case 1 of task 1: (a) tracking trajectory; (b) Cartesian position and velocity errors; (c) contact forces and control laws.

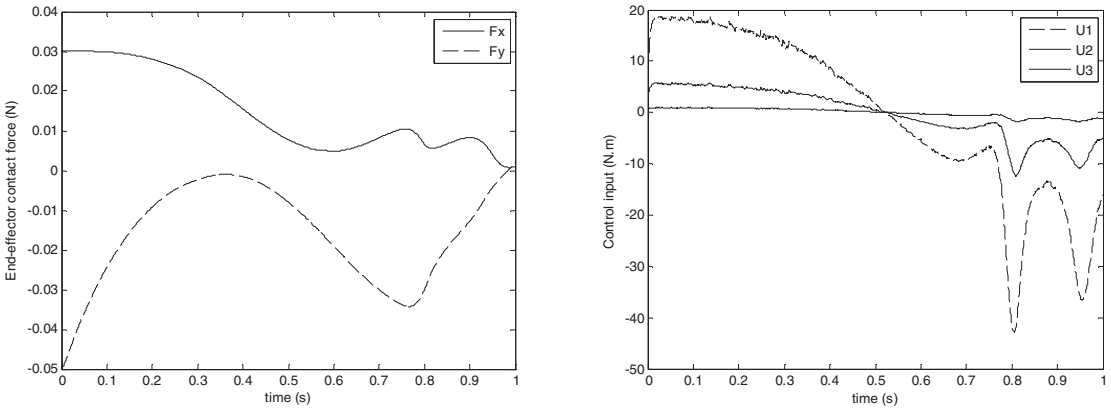
https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press



(a)



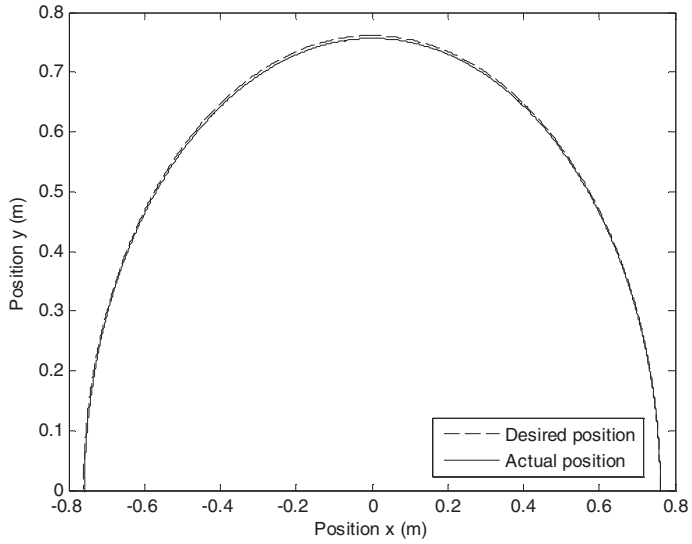
(b)



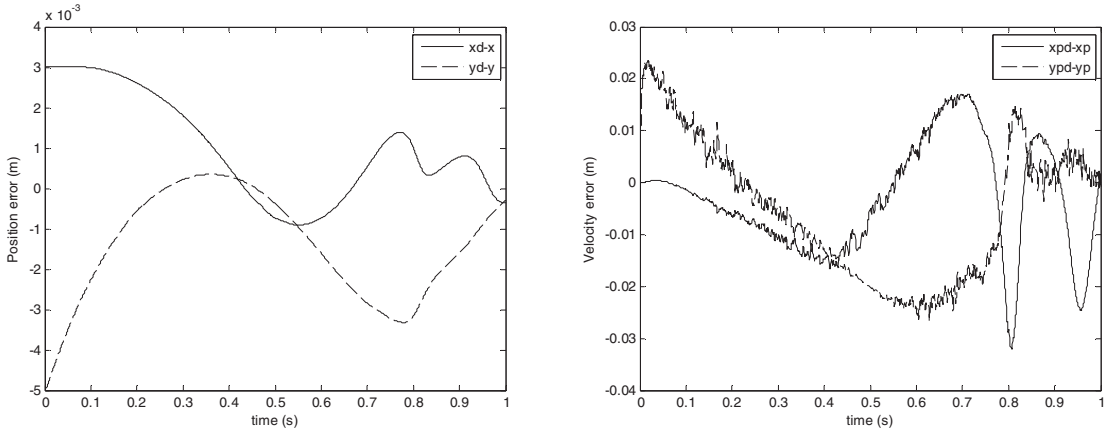
(c)

Fig. 7. Tracking dynamics of the robot for case 2 of task 1: (a) tracking trajectory; (b) Cartesian position and velocity errors; (c) contact forces and control laws.

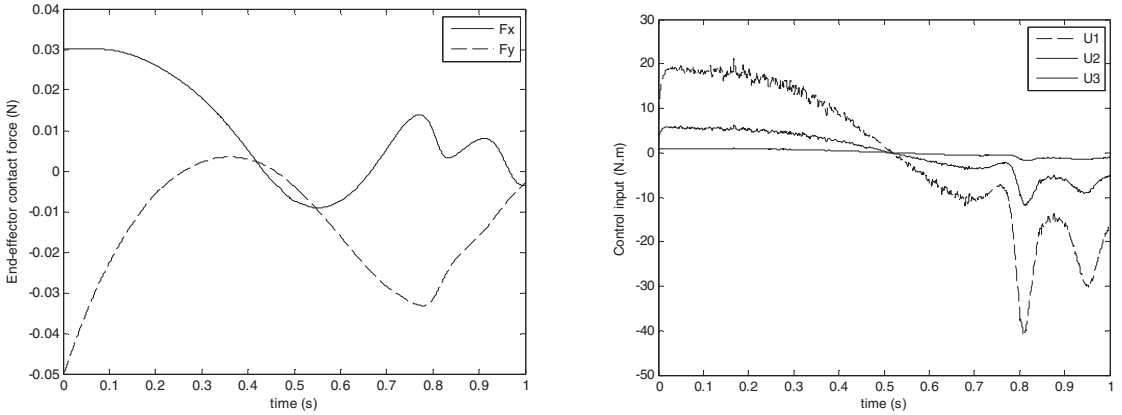
https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press



(a)



(b)



(c)

Fig. 8. Tracking dynamics of the robot for case 3 of task 1: (a) tracking trajectory; (b) Cartesian position and velocity errors; (c) contact forces and control laws.

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

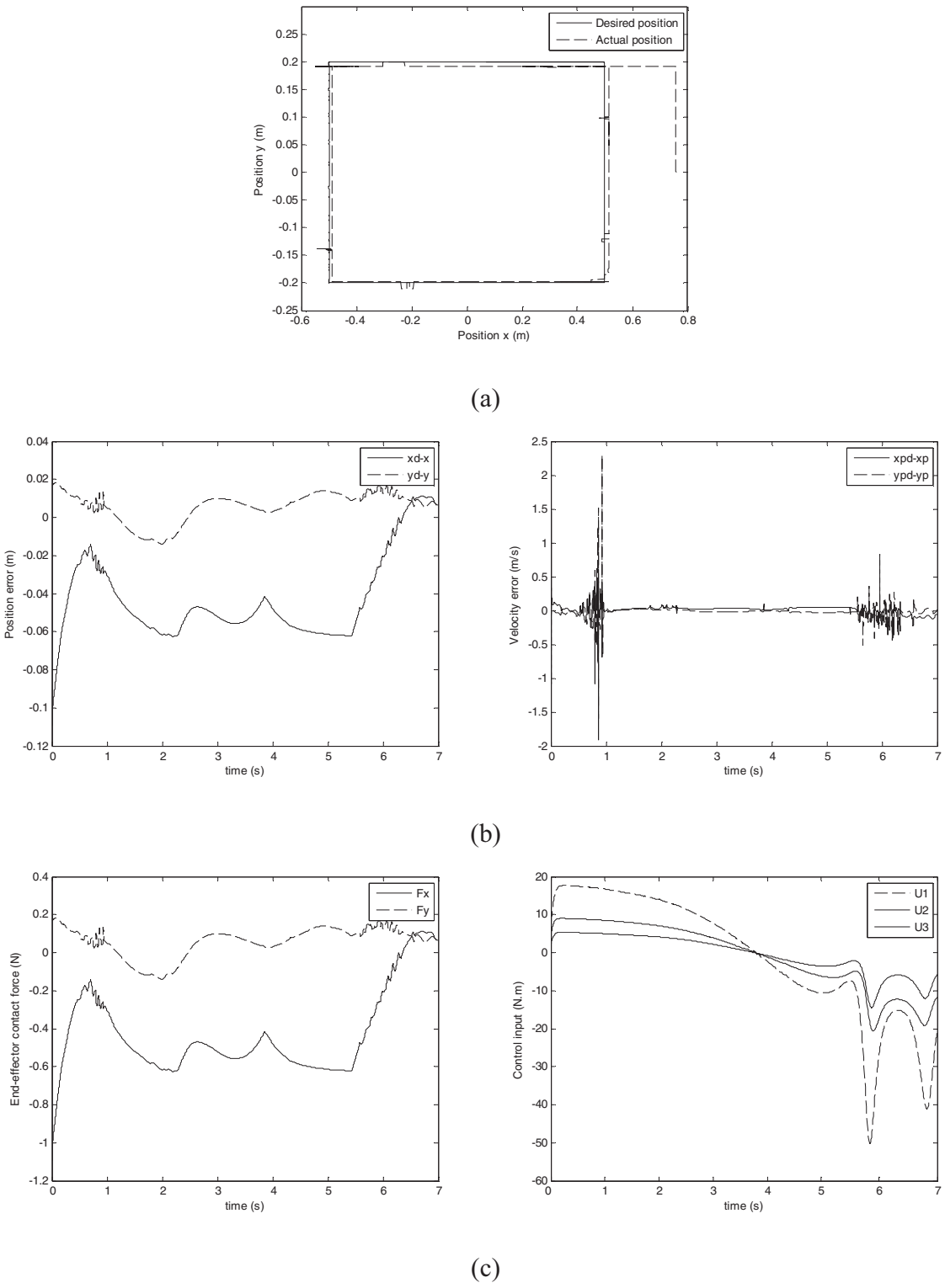


Fig. 9. Tracking dynamics of the robot for case 1 of task 2: (a) tracking trajectory; (b) Cartesian position and velocity errors; (c) contact forces and control laws.

offline, the parameters of the designed controller. The tracking performances of complex trajectories have been shown for a 3-DOF robot-aided rehabilitation device. We have chosen two tracking tasks: a circular profile trajectory and a rectangular one. Extensive simulations have been carried out to verify

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

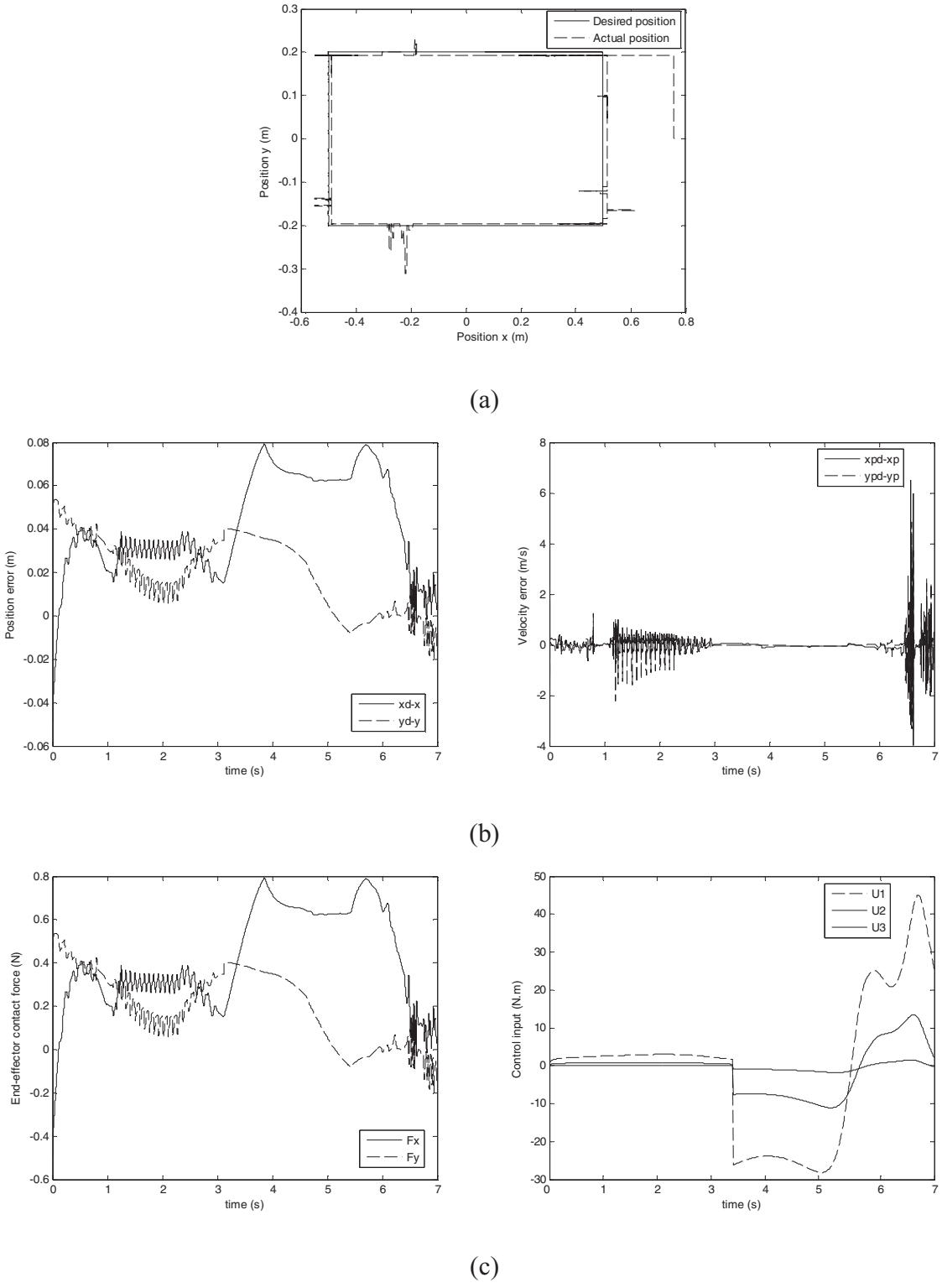


Fig. 10. Tracking dynamics of the robot for case 2 of task 2: (a) tracking trajectory; (b) Cartesian position and velocity errors; (c) contact forces and control laws.

and demonstrate the effectiveness of the proposed scheme. The results obtained have demonstrated the stability and robustness of the scheme in terms of tracking errors, even in the presence of parameter uncertainties and measurement noise.

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

References

1. B. Siciliano and O. Khatib, *Springer Handbook of Robotics* (Springer-Verlag, Berlin, Germany, 2008).
2. B. Siciliano and L. Villani, *Robot Force Control* (Kluwer Academic, Boston, Massachusetts, 1999).
3. G. Carbone, *Grasping in Robotics* (Springer-Verlag, London, UK, 2013).
4. G. Zeng and A. Hemami, "An overview of robot force control," *Robotica* **15**, 473–482 (1997).
5. S. Chiaverini, B. Siciliano and L. Villani, "A survey of robot interaction control schemes with experimental comparison," *IEEE/ASME Trans. Mechatron.* **4**, 273–285 (1999).
6. J. A. Briones, E. Castillo, G. Carbone and M. Ceccarelli, "Position and Force Control of a Parallel Robot Capaman 2 Bis Parallel Robot for Drilling Tasks," *Proceedings of the IEEE International Conference on Electronics, Robotics and Automotive Mechanics (CERMA '09)*, Cuernavaca, Morelos (Sept. 22–25, 2009) pp. 181–186.
7. G. Carbone, E. Villegas and M. Ceccarelli, "Design and validation of force control loops for a parallel manipulator," *Int. J. Intell. Mechatron. Robot.* **1**, 1–18 (2011).
8. Y. Chen, "Parameter fine-tuning for robots," *IEEE Control Syst. Mag.* **9**(2), 35–40 (1989).
9. B. Karan, "Robust position–force control of robot manipulator in contact with linear dynamic environment," *Robotica* **23**, 799–803 (2005).
10. Z. Doulgeri and Y. Karayiannidis, "Force/position control self-tuned to unknown surface slopes using motion variables," *Robotica* **26**, 703–710 (2008).
11. H. Mehdi and O. Boubaker, "New Robust Tracking Control for Safe Constrained Robots Under Unknown Impedance Environment," **In**: *Advances in Autonomous Robotics* (G. Herrmann *et al*., eds.) (Springer-Verlag, Berlin, Germany, 2012) pp. 313–323.
12. H. Mehdi and O. Boubaker, "Robust tracking control for constrained robots," *Procedia Eng.* **41**, 1292–1297 (2012).
13. W. Gueaieb, F. Karray and S. Al-Sharhan, "A robust hybrid intelligent position/force control scheme for cooperative manipulators," *IEEE/ASME Trans. Mechatron.* **12**, 109–125 (2007).
14. B. Achili, B. Daachi, Y. Amirat, A. Ali-Cherif and M. E. Daâchi, "A stable adaptive force/position controller for a C5 parallel robot: a neural network approach," *Robotica* **30**, 1177–1187 (2012).
15. Z. Li, S. S. Ge, M. Adams and W.S. Wijesoma, "Robust adaptive control of uncertain force/motion constrained nonholonomic mobile manipulators," *Automatica* **44**, 776–784 (2008).
16. E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems* (Oxford University Press, New York, 1999).
17. J. Kennedy, R. Eberhart and Y. Shi, *Swarm Intelligence* (Morgan Kaufmann, Burlington, Massachusetts, 2001).
18. M. Clerc and J. Kennedy, "The particle swarm—explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.* **6**, 58–73 (2002).
19. M. Dorigo and M. Birattari, "Swarm intelligence," *Scholarpedia* **2**(9), 1462 (2007).
20. E. Bonabeau, D. W. Corne, J. D. Knowles and R. Poli, "Swarm intelligence theory: a snapshot of the state of the art," *Theor. Comput. Sci.* **411**, 2081–2083 (2010).
21. M. Dorigo, D. Floreano, L. Gambardella, F. Mondada, S. Nolfi, T. Baaboura, *et al*., *Swarmanoid: A Novel Concept for the Study of Heterogeneous Robotic Swarms* (Resource document) (IRIDIA, Université Libre de Bruxelles, Bruxelles, Belgium, 2013). Available at: http://www.idsia.ch/∼gianni/Papers/SwarmanoidPaperTR.pdf (accessed on January 30, 2013).
22. R. C. Eberhart and Y. Shi, "Comparison between genetic algorithms and particle swarm optimization," *Lect. Notes Comput. Sci.* **1447**, 611–618 (1998).
23. M. Dorigo and E. Şahin, "Swarm robotics," *Auton. Robots* **17**, 111–113 (2004).
24. F. Ducatelle, G. Di Caro, C. Pinciroli and L. M. Gambardella, "Self-organized cooperation between robotic swarms," *Swarm Intell.* **5**, 73–96 (2011).
25. S. Nouyan, A. Campo and M. Dorigo, "Path formation in a robot swarm—self organized strategies to find your way home," *Swarm Intell.* **2**, 1–23 (2008).
26. Y. Li, X. Chen, "Mobile robot navigation using particle swarm optimization and adaptive NN," *Lect. Notes Comput. Sci.* **3612**, 554–559 (2005).
27. G. G. Rigatos, "Multi-robot motion planning using swarm intelligence," *Int. J. Adv. Robot. Syst.* **5**, 139–144 (2008).
28. W. Xu, C. Li, B. Liang, Y. Liu and Y. Xu, "The Cartesian path planning of free-floating space robot using particle swarm optimization," *Int. J. Adv. Robot. Syst.* **5**, 301–310 (2008).
29. K. D. Sharma, A. Chatterjee and A. Rakshit, "A PSO–Lyapunov hybrid stable adaptive fuzzy tracking control approach for vision-based robot navigation," *IEEE Trans. Instrum. Meas.* **61**, 1908–1914 (2012).
30. C. Wei-Der and S. Shun-Peng, "PID controller design of nonlinear systems using an improved particle swarm optimization approach," *Commun. Nonlin. Sci. Numer. Simul.* **15**, 3632–3639 (2010).
31. B. Zafer and K. Oguzhan, "A fuzzy logic controller tuned with PSO for 2 DOF robot trajectory control," *Expert Syst. Applic.* **38**, 1017–1031 (2011).
32. H. Mehdi and O. Boubaker, "Impedance controller tuned by particle swarm optimization for robotic arms," *Int. J. Adv. Robot. Syst.* **8**, 93–103 (2011).
33. H. Liu, Y. C. Liu, M. H. Jin, K. Sun and J. Huang, "An experimental study on Cartesian impedance control for a joint torque-based manipulator," *Adv. Robot.* **22**, 1155–1180 (2008).
34. H. Jianbin, X. Zongwu, J. Minghe, J. Zainan and L. Hong "Adaptive impedance-controlled manipulator based on collision detection," *Chin. J. Aeronaut.* **22**, 105–112 (2009).

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press

35. M. Santhakumar and J. Kim, "Indirect adaptive control of an autonomous underwater vehicle-manipulator system for underwater manipulation tasks," *Ocean Eng.* **54**, 233–243 (2012).
36. H. Mehdi and O. Boubaker, "Stiffness and impedance control using Lyapunov theory for robot-aided rehabilitation," *Int. J. Social Robot.* **4**, 107–119 (2012).
37. H. I. Krebs, J. J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Rannekleiv, B. T. Volpe and N. Hogan, "Rehabilitation robotics: performance-based progressive robot-assisted therapy," *Auton. Robots* **15**, 7–20 (2003).
38. R. Loureiro, F. Amirabdollahian, M. Topping, B. Driessen and W. Harwin, "Upper limb robot mediated stroke therapy—GENTLE/s approach," *Auton. Robots* **15**, 35–51 (2003).
39. H. Mehdi and O. Boubaker, "Rehabilitation of a human arm supported by a robotic manipulator: a position/force cooperative control," *J. Comput. Sci.* **6**, 912–919 (2010).
40. R. C. Eberhart and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," *Proceedings of the 6th IEEE International Symposium on Micro Machine and Human Science*, Nagoya, Japan (Oct. 4–6, 1995) pp. 39–43.
41. D. Bratton and J. Kennedy, "Defining a Standard for Particle Swarm Optimization," *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS)*, Honolulu, USA (Apr. 1–5, 2007), pp. 120–127.
42. Y. Shi and R. C. Eberhart, "A Modified Particle Swarm Optimizer," *Proceedings of the IEEE International Conference on Evolutionary Computation*, Anchorage, USA (May 4–9, 1998) pp. 69–73.
43. M. Santhakumar, "Investigation into the dynamics and control of an underwater vehicle-manipulator system," *Model. Simul. Eng.* **2013**, 1–13 (2013).

https://doi.org/10.1017/S0263574714001775 Published online by Cambridge University Press